

Claims

1. (Currently Amended) A computer program product embodied on a first computer-readable medium and comprising code that, when executed, causes a computer to perform a method of generating a partial procedure summary of a procedure of multithreaded software, wherein the procedure performs a plurality of actions when executed, the method comprising:

~~performing during~~ a reachability analysis of at least a portion of the multithreaded software,
reaching the procedure;

when the procedure is reached during the reachability analysis,

1) identifying a at least one plurality of the actions within the procedure as atomically modelable with respect to multithreaded execution of the procedure as atomically modelable actions~~[[;]], wherein the atomically modelable actions are not subject to interruption by other threads,~~ and

2) generating the at least one partial procedure summary of the procedure from the ~~plurality of the~~ atomically modelable actions, wherein the at least one partial procedure summary comprises at least one state pair, wherein the at least one state pair models an initial state and a resulting state of ~~an~~ the atomically modelable actions for the procedure; and ~~the reachability analysis consulting the partial procedure summary to continue the reachability analysis~~

during continuation of the reachability analysis, reaching the procedure a second time; and

wherein if initial state of the procedure reached the second time is the same as the initial state

of the at least one partial procedure summary, then the resulting state is used as procedure state in the reachability analysis.

2. (Cancelled)

3. (Previously Presented) The computer program product of claim 1, the method further comprising:

modeling execution of the software such that the state pair that comprises the partial procedure summary is executed rather than the atomically modelable actions.

4. (Previously Presented) The computer program product of claim 1, the method further comprising:

during modeling, comparing an indicated state invariant with a modeled state; and
responsive to determining the modeled state fails the indicated state invariant, indicating that a programming flaw is present in the software; wherein determining the modeled state fails the indicated state invariant comprises determining that a condition is false for at least one execution path.

5. (Previously Presented) The computer program product of claim 1, wherein the resulting state comprises at least one of a plurality of possible states of the multithreaded software after execution of the modeled procedure, the method further comprising:

storing an initial program counter location within the modeled procedure for the initial state;
storing a resulting program counter location within the modeled procedure for the resulting

state; and

associating the initial program counter location within the modeled procedure and the resulting program counter location within the modeled procedure with the partial procedure summary.

6. (Currently Amended) The computer program product of claim 1

wherein the reachability analysis consulting the at least one partial procedure summary further comprises determining possible execution paths within the procedure and using the procedure summary to explore possible states.

7. (Previously Presented) The computer program product of claim 1 wherein the

identifying comprises identifying a transaction boundary within the actions.

8. (Previously Presented) The computer program product of claim 1 wherein the

identifying comprises identifying at least one of the plurality of actions as movable later in time with respect to actions executed by other threads without affecting a resulting end state.

9. (Previously Presented) The computer program product of claim 1 wherein the

identifying comprises identifying a sequence of actions having zero or more right movers followed by an atomic action followed by zero or more left movers.

10. (Previously Presented) The computer program product of claim 1 wherein the plurality

of actions atomically modelable with respect to multithreaded execution of the software is a subset of the plurality of actions of the procedure, the subset comprising less than all of the plurality of actions

of the procedure.

11. (Currently Amended) A computer program product embodied on a first computer-readable medium and comprising code that, when executed, causes a computer to perform a method of modeling multithreaded software, the method comprising:

performing a reachability analysis of the multithreaded software;

during the reachability analysis, reaching a procedure;

analyzing actions of the multithreaded software within the procedure such that actions that can be executable atomically within the procedure are determined, wherein actions that can be executable atomically are not subject to interruption by other threads;

based on the analyzing, generating a plurality of partial procedure summaries for the multithreaded software, the plurality of partial procedure summaries comprising respective start and end actions for the determined actions executable atomically; and

during the reachability analysis, again reaching the procedure and reusing the plurality of partial procedure summaries to determine actions executable atomically;

wherein the partial procedure summaries comprises a plurality of modeled states of the multithreaded software for multithreaded execution of the multithreaded software.

12. (Currently Amended) The computer program product of claim 11 wherein at least one of the partial procedure summaries comprises at least two or more partial procedure summaries summarizing a procedure.

13. (Currently Amended) The computer program product of claim 11 wherein at least one

of the partial procedure summaries comprises at least one partial procedure summary for a procedure, wherein the partial procedure summary summarizes less than all of the procedure.

14. (Currently Amended) The computer program product of claim 11 wherein the analyzing comprises:

identifying a series of transactions within the multithreaded software; and
modeling the transactions via the partial procedure summaries.

15. (Currently Amended) A computer-implemented system for modeling multithreaded software, the system comprising:

a model checker operable to analyze a model of the multithreaded software via checking the model of the multithreaded software for programming flaws, the model checker comprising:

the model of the multithreaded software,

wherein the model comprises:

a plurality of partial procedure summaries modeling beginning states and ending states of partial summaries of procedures within the multithreaded software during multithreaded execution of the multithreaded software, the partial procedure summaries comprising the start and end states of sets of actions within the procedures, the actions atomically modelable with respect to multithreaded execution of the software in that the atomically modelable actions will all be performed within a single procedure by a same thread; and

a reachability analyzer operable to employ the partial procedure summaries instead of the sets of actions to generate modeled states of the software.

16. (Currently Amended) The computer-implemented system of claim 15 wherein at least one of the partial procedure summaries comprises a partial procedure summary summarizing actions deemed to have occurred one after another without interruption by another thread.

17. (Canceled)

18. (Previously Presented) The computer-implemented system of claim 16 wherein the system is operable to detect programming flaws via comparing an indicated state invariant with the modeled states.

19. (Cancelled)

20. (Cancelled)

21. (Previously Presented) The computer program product of claim 1, wherein the at least one state pair comprises an initial state of the procedure and at least one of a plurality of possible states of the multithreaded software after execution of the modeled procedure.

22. (Canceled)

23. (Currently Amended) The computer program product of claim 1, wherein the resulting state is in a location in the procedure prior to the location of the initial state in the procedure.

24. (Previously Presented) The computer program product of claim 1, wherein the partial procedure summary of the procedure of multithreaded software is embodied on a second computer readable medium.